Ace 1.1 User's Manual

by
Kendra Z. Hoff, Sara E. Ort, Bruce L. Rives, and Kirk D. McGraw

Computer-aided drafting systems and analysis programs
have become common tools in Army facility design, but
these technologies have been developed and transferred
in a piecemeal fashion. While these tools successfully
optimize specific design processes such as facility layout,
energy analysis, and structural analysis, they essentially
comprise "islands of automation" that cannot successfully
interact. Resulting problems include the need to
redundantly enter data and an overload of data not
applicable to all design tasks. Previous attempts to
integrate these automated tools have been expensive,
and have met with limited success.

To address this problem the U.S. Army Construction
Engineering Research Laboratories (USACERL)
developed Ace, an integrated design-assistant technology
that automates routine design tasks and incorporates key
project information into a central database for access by
all participants in various design and specification
processes. Ace uses heuristic rules, database
technology, and a powerful checklist facility to improve
productivity and ensure repeatable design quality.

This report documents the operation of Ace version 1.1.
System requirements for Ace 1.1 are (1) a 33 MHz Intel
80386 microprocessor or equivalent, (2) 16 Mb of random
access memory (RAM), (3) a 3.5 in. high-density floppy
drive, and (4) the Microsoft Windows® 3.1 or Windows
NT® 3.5 operating system.

19960311 186

DTIC QUALITY INSPECTED 5

DISCLAIMER NOTICE

UNCLASSIFIED

Technical Report
distributed by

DEFENSE
TECHNICAL
INFORMATION
CENTER

DTIC | Acquiring Information-
Imparting Knowledge

Cameron Station
Alexandria, Virginia 22304-6145

UNCLASSIFIED

THIS DOCUMENT IS BEST
QUALITY AVAILABLE. THE
COPY FURNISHED TO DTIC
CONTAINED A SIGNIFICANT
NUMBER OF PAGES WHICH DO
NOT REPRODUCE LEGIBLY.

USER EVALUATION OF REPORT

REFERENCE: USACERL ADP Report 95/54, *Ace 1.1 User's Manual*

Please take a few minutes to answer the questions below, tear out this sheet, and return it to USACERL. As user of this report, your customer comments will provide USACERL with information essential for improving future reports.

1. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

_____

_____

_____

2. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.)

_____

_____

3. Has the information in this report led to any quantitative savings as far as manhours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

_____

_____

4. What is your evaluation of this report in the following areas?

a. Presentation: _____

b. Completeness: _____

c. Easy to Understand: _____

d. Easy to Implement: _____

e. Adequate Reference Material: _____

f. Relates to Area of Interest: _____

g. Did the report meet your expectations? _____

h. Does the report raise unanswered questions? _____

i. General Comments. (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.)

_____

_____

_____

_____

_____

_____

5. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: _____

Telephone Number: _____

Organization Address: _____

_____

_____

6. Please mail the completed form to:

Department of the Army
CONSTRUCTION ENGINEERING RESEARCH LABORATORIES
ATTN: CECER-TR-I
P.O. Box 9005
Champaign, IL 61826-9005

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE June 1995	3. REPORT TYPE AND DATES COVERED Final

4. TITLE AND SUBTITLE
Ace 1.1 User's Manual

5. FUNDING NUMBERS
4A162784
AT45
FE-X55

6. AUTHOR(S)
Kendra Z. Hoff, Sara E. Ort, Bruce L. Rives, and Kirk D. McGraw

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Construction Engineering Research Laboratories (USACERL)
P.O. Box 9005
Champaign, IL 61826-9005

8. PERFORMING ORGANIZATION REPORT NUMBER
ADP 95/54

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Headquarters, U.S. Army Corps of Engineers
ATTN: CEMP-ET
20 Massachusetts Ave NW
Washington, DC 20314-1000

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Copies are available from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)
Computer-aided drafting systems and analysis programs have become common tools in Army facility design, but these technologies have been developed and transferred in a piecemeal fashion. While these tools successfully optimize specific design processes such as facility layout, energy analysis, and structural analysis, they essentially comprise "islands of automation" that cannot successfully interact. Resulting problems include the need to redundantly enter data and an overload of data not applicable to all design tasks. Previous attempts to integrate these automated tools have been expensive, and have met with limited success.

To address this problem the U.S. Army Construction Engineering Research Laboratories (USACERL) developed Ace, an integrated design-assistant technology that automates routine design tasks and incorporates key project information into a central database for access by all participants in various design and specification processes. Ace uses heuristic rules, database technology, and a powerful checklist facility to improve productivity and ensure repeatable design quality.

This report documents the operation of Ace version 1.1. System requirements for Ace 1.1 are (1) a 33 MHz Intel 80386 microprocessor or equivalent, (2) 16 Mb of random access memory (RAM), (3) a 3.5 in. high-density floppy drive, and (4) the Microsoft Windows 3.1 or Windows NT 3.5 operating system.

14. SUBJECT TERMS		15. NUMBER OF PAGES 52
computer-aided design facility design user manuals	integrated software	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

Foreword

Contents

Chapter 6: The Instance View   35

Chapter 1: Introduction

Background

The quality and life-cycle cost of military facilities is a crucial issue for the U.S. Army and the other military services. Facilities have a great impact on military readiness and deployability, personnel recruitment and retention, and training. Military facilities that cost too much to design, build, maintain, and operate--whether in terms of dollars, time, or human resources--translate into problems for the national defense.

Engineers and architects working on projects for the Department of Defense (DoD) face unusual challenges. They are subject to rules and regulations above and beyond those that govern them in the private sector. Examples include complicated bid specification formats, energy policy requirements, etc. Furthermore, they are routinely required to design buildings of diverse types and functions, from family housing on base to desert refugee camps to hangers for low-temperature testing of jet aircraft.

In recent years, facility designers have in their daily work been relying more and more heavily on computer-aided drafting (CAD) systems and analysis programs. These technologies have been developed and transferred in a piecemeal fashion, however focusing on the optimization of specific design processes such as facility layout, energy analysis, and structural analysis. The results have been "islands of automation" that create an entire new set of problems, including information overload, redundant data entry, and inconsistent design and construction specification documents. Previous attempts to integrate these tools have been expensive, and have met with limited success.

To address this systemic problem, the U.S. Army Construction Engineering Research Laboratories (USACERL) has developed Ace, an integrated design-assistant technology capable of automating routine design tasks and incorporating key project information into a central database for ready access by all participants in various design and specification processes. Ace uses heuristic rules, database technology, and a powerful checklist facility to enhance productivity and ensure repeatable design quality.

Purpose

The purpose of Ace is to provide a computer operating environment that will enable facility design professionals to use data from CAD drawings, analysis programs, and bid specifications to accurately execute various design processes while minimizing or eliminating the need for redundant data input.

System Requirements

The minimum hardware and software requirements for Ace 1.1 are:

- a 33 MHz Intel 80386 microprocessor or equivalent

- 16 Mb of random access memory (RAM)

- Microsoft Windows® 3.1 operating environment or Windows NT® 3.5 operating system

- 3.5 in. high-density floppy drive.

This manual assumes that the user is familiar with basic terminology and operating concepts of microcomputers and the Windows® environment.

Mode of Technology Transfer

Ace is distributed, supported, and maintained by the Engineering Processes Division of the Planning and Management Laboratory (CECER-PL-E), USACERL. Queries may be directed to the Engineering Processes Division at 1-800-872-2375, extension 7357; 217-373-6724 (fax); or by mail at USACERL, ATTN: CECER-PL-E, PO Box 9005, Champaign, IL 61826-9005.

Chapter 2: The Ace Environment

Ace Overview

Ace is a software environment targeted for use in the architectural and engineering design domains. The application of Ace can be tailored to match a specific design process. As a result, the functionality of the system can vary for different users. Ace can be used to:

- Integrate design automation tools such as CAD systems, bid specifications, databases, and analysis programs through shared information.

- Provide consistency throughout documents within a project by automatically performing operations if any information changes.

- Facilitate consistent decision making across projects and preserve design rationale.

- Determine the appropriate information to feed a program, execute the program by imitating an expert user, and then review the results.

- Automate mundane and repetitive tasks.

- Accept incomplete data and make decisions based on rules-of-thumb.

- Check designs and provide notification of possible errors or unreasonable situations.

- Verify compliance with policies, codes, and regulations.

- Model the design process and track the completion of tasks.

Ace excels as an integration platform by working with databases, CAD packages, analysis programs, and design specifications. Ace also excels as a modeling tool. It allows users to interact with "real" objects such as pumps and walls instead of arcs, lines, and data arrays. Relationships such as "Part Of" help produce a rich model of the design. This model can be operated on by heuristic rules and bi-directional constraints. Ace calls this model a project, as it corresponds to a real-world project in which the goal is to define some engineered object or system.

Applications written for Ace, called *agents*, make use of these capabilities to assist the user in building and refining the design model. For routine design, agents may be organized into a *process*, which corresponds to an engineering process used by

an individual or organization. The user is able to control the operation of agents through their *checklists*, which may be thought of as a list of tasks that the agent helps the user perform.

The Ace program is used in two different ways. Ace is used as a tool for the project, with processes, agents, and checklists, as describe above. Ace is also used to develop the processes, agents, and checklists. To use Ace as a tool for your project, you work in the *Project window*. To develop Ace components, you work in the *Developer window*. This manual documents the Project window. The Developer's Manual documents the Developer window.

There also is a third window, the *LISP window*. Ace has been developed using the LISP programming language. When unexpected programming errors occur, a message will appear in the LISP window.

Ace Components

Every action taken in a project is performed by an *agent*. Agents are expert systems that are tightly integrated with traditional CAD tools and other engineering applications. ACE agents communicate with each other using libraries of design objects such as walls, fans, or pumps. Agent-based systems are well suited for use as integration platforms. Complex tasks can gradually be implemented using many small agents. The resulting system is easy to maintain since an agent may be modified or replaced without affecting the remaining agents. Although most agents act under the user's direction, they can run in the background and act in an advisory capacity. Such an agent might use rule-based techniques to check for building code violations or act as a source of expertise by making suggestions on how to improve energy efficiency.

An agent may contain one or more checklists. Checklists are a way to organize the tasks for the agent. They allow the automation of routine design tasks.

Agents can be organized into processes. Processes model a specific design process by specifying which agents and checklists are used to perform a sequence of operations. A process can handle large, routine design problems, specifying how an agent will use its checklists.

The data must be organized to be useful. The set of *frames* creates a description of the data structure. Each frame describes an entity such as the walls, windows, ductwork, spaces, and construction requirements. Attributes of frames, called *slots*, describe the characteristics of the frame. The slots for a wall frame could include length, height, and construction material. Slots for a construction requirement frame might include descriptions, and governing regulations.

Instances store project data. Instances, combined with constraints and relationships, called semantic links form the design model. An instance is created using a frame as a template such that any slots defined by the frame become data fields in the instance.

In the example below, the Ace project uses three agents: a Loads Agent, a Routing Agent, and a Cost Agent. The Loads and Routing Agents are part of the Hydronic Systems Process. The Cost Agent is not part of a process for this project. It works alone after the Hydronic Systems Process has completed.
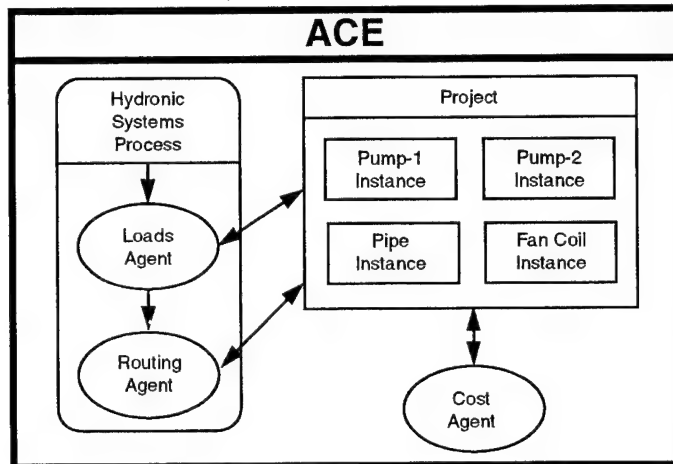
Example of Ace components

Checklists within the Loads Agent have created two instances of a Pump Frame and one each of a Pipe Frame and a Fan Coil Frame. The instances contain the data for the project. The Loads Agent provides some of the data. The Routing Agent uses some of the data and adds some new data. The Cost Agent uses data from other slots of the instances. Flow-rate and power are two possible slots in the Pump Frame.

The Ace User's Manual is designed to assist the person that wants to use existing agents, processes and frames. The Ace Developer's Manual is to be used by those creating agents, processes and frames.

Agents

An agent is a software entity that functions as an independent source of expertise designed to perform a particular task or set of tasks. When an agent is added to a project, it is similar to bringing a specialist with specific skills into a room to work on the project.
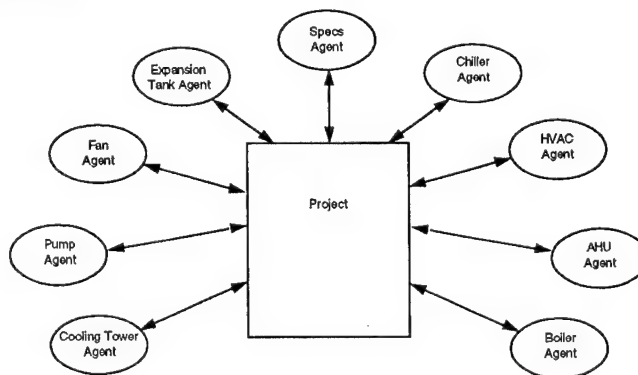
Example of agents in a mechanical design office

The preceding figure shows a set of agents used by a mechanical design office to develop a set of contract specifications for heating, ventilating, and air conditioning (HVAC) equipment. Each agent performs a specific task for the engineer. For example, the Chiller agent gathers information about the required cooling load from the user. The agent then uses a Chiller sizing program to determine an appropriate size.

Each agent is unique and can have several characteristics. Agents can:

- Automate routine design procedures which are executed through the use of checklists.

- Monitor a process and perform an action when certain conditions are met. For example, a fire safety agent can monitor the project for the occurrence of doors that provide access to stairwells and check their fire rating to ensure that the stairwells are safe; the agent can bring unsafe fire doors to the user's attention immediately.

- Execute existing software programs and integrate them into Ace. For example, an energy agent can control a powerful energy analysis software tool called BLAST (Building Loads Analysis and System Thermodynamics). The agent can determine the appropriate information to feed the analysis, run the analysis, and review the results.

- Create and modify instances, which store the project information.

Agents are used in projects through the Ace Project window. They are created through the Developer window. For information on using agents, see "Using the Process View" on page 19, and "Using the Agent View" on page 27. For information on creating new agents, refer to the Ace Developer's Manual or the Help menu in the Developer window.

Checklists

Each checklist consists of a hierarchical list of items. Some of these items are required and some may be optional, to be done under certain conditions or at the user's discretion. The required items must be completed before the checklist can be considered complete. The order of the items on a checklist reflects the order in which the items should be completed, as shown in the sample that follows:

PUMP Checklist
- Size pumps for solar collector
    - Determine collector flow
        - Determine type of collector
        - Count collectors
    - Determine head requirements
        - Highest point of collector
        - Pressure loss in pipe
    - Determine duty cycle
    - Size pump
        - Analytical calculation
        - Catalog lookup

A simple checklist for sizing pumps

For information on using checklists, see "Using the Process View" on page 19 and "Using the Agent View" on page 27. For information on creating new checklists, refer to the Ace Developer's Manual or the Help menu in the Developer window.

Processes

Predefined processes can be used to easily add the desired agents and checklists to a project. Processes track the status of agents and checklists within the process and act as a flow manager when executed by determining which checklist needs to be completed next.
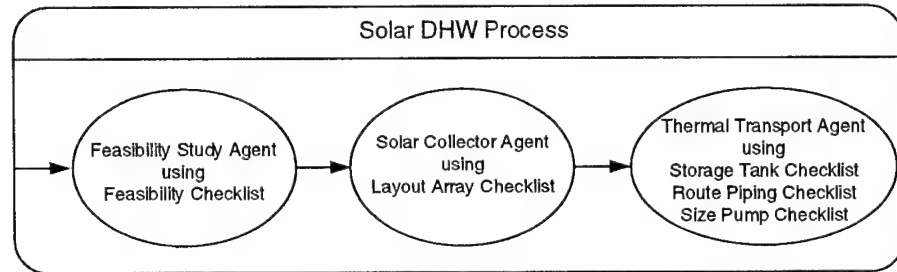
A process for solar domestic hot water (DHW)

Processes are used in projects through the Ace Project window and created through the Developer window. For information on using processes, see "Using the Process View" on page 19. For information on creating new processes, refer to the Ace Developer's Manual or the Help menu in the Developer window.

Frames and Instances

Frames have a hierarchy structure where one frame can inherit slots from another frame. This is analogous to a child inheriting characteristics from a parent; the frames are referred to as parent frames and child frames. There is no limit to the levels in the hierarchy. A child frame will contain specified slots of its parent frame as well as some unique to itself. A parent frame can have many child frames, and likewise a child frame can have many parent frames. All frames in Ace are child frames of a root frame called BB-OBJECT.

Frames and slots have names, which are carried over to the instances. For example, a frame describing a pump may be named "Pump" while two of its slots may be named "flow" and "power", as shown below.
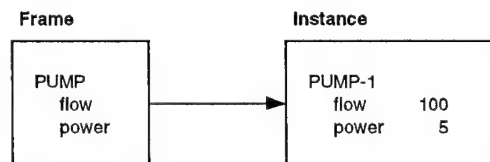
A simple pump frame and pump instance

The figure shows a simplified example of a pump frame and one of it's instances. The frame has slots for the flow of the pump and power. The frame represents a template for pump instances. The instance, called "pump-1", represents an actual pump in the project. The values of the slots in the instance are filled with actual values.

Frames are created in the Developer window and stored in a *library*. For information on creating frames and libraries, refer to the Ace Developer's Manual or the Help menu in the Developer window. Instances are created within agents, including the User Agent. For information on using instances, see "Using the Agent View" on page 27 and "Using the Instance View" on page 35.

Chapter 3: Projects

Using Projects

When you first start Ace, the Project View is activated. You can begin working by creating a new project or by opening an existing project. A project in Ace corresponds to your design project. The project must be saved to keep any work you do. Only one project can be active at a time, and it must be closed before opening a different project or leaving Ace. If you have no need for a project, you may delete the project completely.
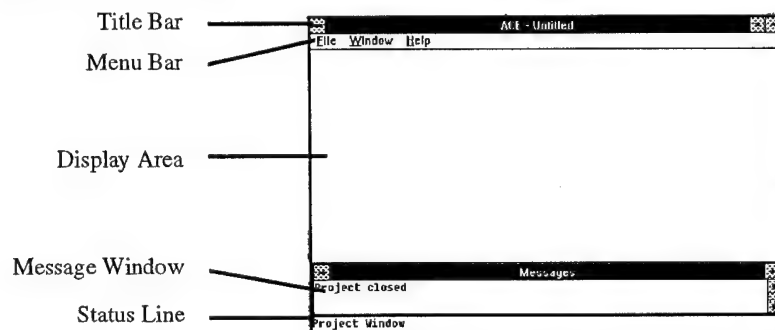
The Project Window contains the following:

- The *Project title bar* displays "Ace" and the name of the active project.

- The *Project menu bar* contains the available menus that can be opened to choose commands.

- The *display area* is used to display the selected project view for the active project.

- The *Message window* displays messages about commands being performed.

- The *Status line* displays the name of the window that is currently under the mouse pointer.

For a quick reference to the available project options, see "Summary of Project Options" on page 18.

Accessing a Project

New projects can be created and existing projects can be accessed at any time during your Ace session. If you are currently working in a project, you will need to save that project and close it before beginning work in another project. You can do this using the menu options or Ace will ask if you want to save the project before it is closed.

Sometimes you will want to begin a project similar to an existing one. Instead of creating a new project and duplicating the work you did for the first project, you can save the existing project with a new filename, then make the changes required for the new project.

To create a new project

1. Choose New from the File menu.

   If a project is currently open, you will be asked if you want to save it before it is closed.

2. Enter a unique filename for new project.

   The filename must contain up to 8 characters with no extension. Ace will automatically add the extension *.prj*.

3. Enter the requested information in the Project Information dialog box.

   This description is used to help the user select the desired project. For information on viewing or modifying these project properties after they have been entered, see "Accessing the Properties of a Project" on page 15.

4. Add any processes, agents, and instances needed for this project.

   Choose the Process View or the Agent View from the View menu. For information on the Process View, see "Using the Process View" on page 19. For information on the Agent View, see "Using the Agent View" on page 27.

To create a new project based on an existing one

1. Open the desired existing project.

2. Choose Save As from the File menu.

3. Replace the filename with a unique name of up to 8 characters.

   The new filename will appear in the title bar for the project. After you assign a new filename, the project is ready for your new changes.

To open an existing project

An existing project can be opened after first launching Ace or after working with other projects.

1. Choose Open from the File menu.

   If a project is currently open, you are asked if you want to save it before it is closed.

2. Select the project you wish to open form the list that appears.

Accessing the Properties of a Project

The properties of a project are for identifying information required by your organization. The fields provided are designed to be universal, but you may add or delete fields to customize them for your group. CAUTION: changing fields affects all existing projects. Existing projects may experience difficulty when opened after changes. For information on how to modify the project properties fields, refer to the Ace Developer's Manual or the Help menu in the Developer window.

Properties fields include: description, start date, CAD system, CAD command line, CAD drawing directory, project number, designer, city, state, and installation. The filename and version number fields are used by Ace and cannot be changed.

To access the project properties

1. Choose Properties from the File menu.

2. Modify the displayed information as needed.

   The filename and Ace version number cannot be changed.

Viewing a Project

There are three ways to examine and interact with your project: (1) the Process View, (2) the Agent View, and (3) Instance View.

- The Process View shows all of the processes used in your project, the agents and checklists contained in each process, and the order of their use. You can execute the checklists in the Process View, and add new processes also. For more information, see "Using the Process View" on page 19.

- The Agent View shows all agents in a project with their checklists and instances. You can add new agents to your project. You can also execute checklists, and view and modify the data in the instances created by each agent. For more information, see "Using the Agent View" on page 27.

- The Instance View allows greater control over instances. You can create and delete them, change the data in them, and create relationships between instances. For more information, see "Using the Instance View" on page 35.

When you open a project for the first time in an Ace session, you will be placed in the Process View. Ace will save the current view when you close that project. If the project is reopened during the same Ace session, the last view in use will be displayed.

Saving a Project

To keep any new information or changes in your project, you must save the project. It is a good idea to save your project after any changes. When you exit the project by opening another project, creating a new project, or exiting the program, Ace will prompt you to save the current project before closing it. If you save without exiting the project, you will return to the window in which you were working.

To save the current project

- Choose Save from the File menu.

Deleting a Project

When you have no more need for a project, it may be deleted. Most of the time you will want to keep the project on hand to refer to it or to use for another project. However, the delete can be used for projects that have major mistakes or ones that are very old and no longer useful.

To delete a project

- Choose Delete from the File menu.

  Select the project from the list.

Closing a Project

You must close the current project before opening another project, creating a new project, or ending an Ace session. Before closing a project, Ace will prompt you to save the project.

To close a project

- Choose Close from the File menu.

  You will be asked if you want to save the project before it is closed.

Accessing the Developer Window

The Developer window can be opened while working with projects. You have to be in the Developer window to install agents. Once the window is open, you can move between the Developer and Project windows by selecting the one you want. For information on using the Developer window, refer to the Ace Developer's Manual or the Help menu in the Developer window.

- Choose Developer from the Window menu to open the Developer window.

Accessing the LISP Window

The LISP (List Processing) window can be opened while working with projects. If your project suddenly does not respond, check the LISP window for programming errors. This should not be necessary when working with a project. Once the window is open, you can move between the LISP and Project windows by selecting the one you want.

- Choose LISP from the Window menu to open the LISP window.

Installing and Uninstalling Agents

Before any agent can be used in Projects, the agent must be installed. Installing loads the agent into memory, making it available to the project. The agent needs to be loaded only once. After an agent has been installed, it will be loaded whenever Ace is started.

When agents installed in memory are no longer needed, you can remove them from memory by uninstalling.

To install an agent

1. Activate the Developer window.

2. Choose Install Agent from the File menu.

   A standard Windows file selection window will pop up.

3. Select the file containing the agent to install.

   A message will appear in the Message window, stating that the agent was installed. If the agent was previously installed, a message will state that.

To uninstall an agent

1. Activate the Developer window.

2. Choose Uninstall Agent from the File menu.

   A popup window will appear containing a list of the installed agents.

3. Select the agent to remove.

   A message will appear in the Message window stating that the agent was uninstalled.

Exiting Ace

Exiting will end your Ace session.

- Choose Exit from the File menu.

or

- Double-click on the Control-menu box in the upper-left corner.

Summary of Project Options

The Project window menu bar contains File, Window, and Help selections. File is for working with the projects. Window allows you to switch to the Developer window and the LISP window, and to organize the open windows. Help provides access to the Ace online help. The major submenus under each main menu are shown below:

File

New	Initiates a new project.
Open	Opens an existing project.
Close	Closes the current project.
Save	Saves changes made in the current project.
Save As	Saves the current project under a new filename.
Delete	Deletes a selected project.
Properties	Displays and allows modification of the project information for the current project.
Exit	Exits Ace.

Window

Project	Activates the Project window.
Developer	Opens and activates the Developer window.
Cascade	Arranges all windows to overlap, with each window title bar visible.
Tile	Changes window sizes so all active windows are equal sizes and arranged neatly on screen.
LISP	Opens and activates the LISP window.
Messages	Opens and activates the Message window.

Help

Contents	Displays a list of Help topics available for Ace.
Search for Help on...	Displays a dialog box for locating information about a specific Ace topic.
How to Use Help	Displays guidance on using Help.
Overview	Displays the Ace Environment topic in Help.
Project Help	Displays the "Using Projects" topic in Help .

Chapter 4: The Process View

Using the Process View

As a grouping of agents and selected checklists that perform a sequence of operations, a process is meant to model a design process.   While an agent is created to do specific tasks, a process may require some of those tasks but not all of them. Thus several processes can contain the same agent, but use different checklists from that agent. When executed, a process acts as a flow manager and specifies which agents and checklists will work on the project, and in what order. For an overview of processes, see "Processes" on page 11.

The Process View allows you to add and remove processes in a project, view the project status, and execute tasks. Within the Process View, the Process window displays a tree of all processes in the project, and the Checklist window displays checklists.   Checklists can be executed in the Checklist window.
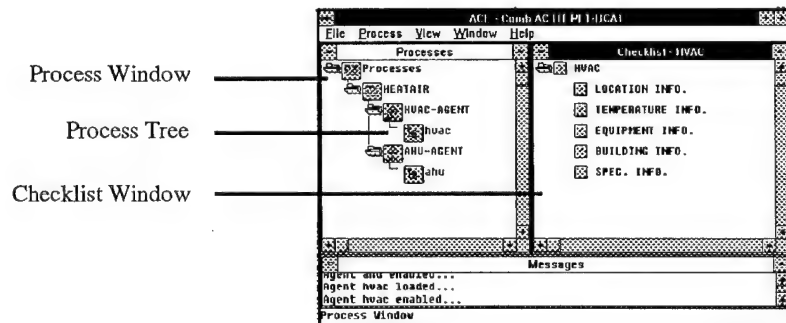
The Process View contains the following:

- The *Project title bar* displays "Ace" and the name of the active project.

- The *Process View menu bar* contains the available menus that can be opened to choose commands.

- The *Process window* displays processes and their agents and checklist.

- The *Checklist window* displays a selected checklist.

- The *Message window* displays messages about commands being performed.

- The *Status line* displays the name of the window that is currently under the mouse pointer.

For a quick reference to the options available, see "Summary of Process View Options" on page 23.

To access the Process View

- Choose Processes from the View menu.

  The Process View will appear in the Project window.

Viewing and Modifying the Process Tree

The Process window displays the process tree. The process tree is a hierarchical structure with all processes shown on the same level. Agents are shown under the process in which they are included, and the selected checklists are shown under their agent. This tree contains the processes in the current project in logical order of execution. Processes can be added to or removed from the process tree. Agents and checklists are displayed in the tree as well.

The process tree contains icons and buttons that can be executed.

- By selecting *folder icons*, the tree can be expanded to show the agents and checklists included in a process.

- The *process*, *agent*, and *checklist buttons* enable you to view the checklist items that need to be completed.
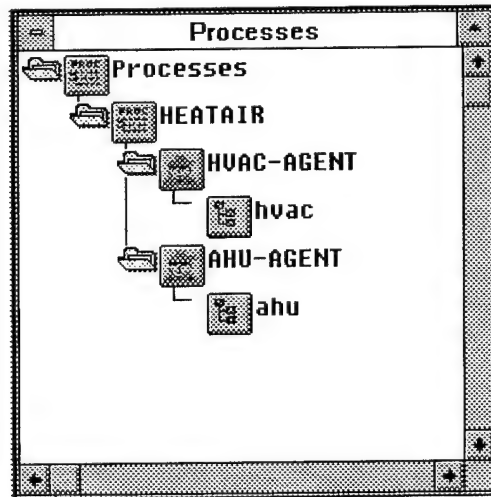
To display the components of the process tree

The folder icons are used to display or hide the items on the process tree. A folder icon appears open if an item is expanded and closed if an item is collapsed. An item without a folder icon next to it has no items below it in the hierarchy.

- Clicking on the folder icon next to the button labeled Processes, at the top of the tree, displays or hides the processes' names in the current project.

- Clicking on the folder icon next to a process button displays the process' agents names, or hides them.

- Clicking on the folder icon next to an agent button displays or hides the agent's checklists names.

To add a process to a project

1. Choose Add from the Process menu.

   A dialog box will appear.

2. Select the file that contains the process you want to add.

   The process will be added to the process tree.

   The default extension for process files is *.pcs*. The dialog box will display the Ace default process directory. For information on Ace configuration files, refer to the Ace Developer's Manual or the Help menu in the Developer window.

To remove a process from a project

1. Choose Remove from the Process menu.

   A dialog box will appear containing a list of the processes in the project.

2. Choose the process you want to remove.

   The process will be deleted from the process tree.

Executing Checklist Items in the Process View

The user may operate in any desired order, however, the normal execution of a project is top-down, i.e., the first process on the tree is executed, then the next, continuing to the last. Within each process, each agent is executed in order. For each agent, the checklists are completed top-down also. Checklists are displayed in the Checklist window as a hierarchical tree of items. Each item can be then be executed to supply a specific piece of information or perform a certain function. For an overview of checklists, see "Checklists" on page 10.

The process tree contains process, agent, and checklist buttons. Selecting one of these buttons with the mouse displays a checklist in the Checklist window as shown below.

- Pressing a *process* button will display the first checklist that has not been completed. The agents will be examined in order. If all checklists in all of the agents have been completed, a message to that effect will be displayed and the Checklist window will not change.

- Pressing an *agent* button will display the first checklist in the agent that has not been completed. If all required checklists in that agent have been completed, a message will be displayed and the Checklist window will not change.

- Pressing a *checklist* button will display the checklist directly.

```
┌─────────────────────────────────────────────────┐
│ ▣         Checklist - HVAC              ▲        │
├─────────────────────────────────────────┬───────┤
│ 🖛▦  HVAC                                │ ▲     │
│                                         │ ▒     │
│       ▦  LOCATION INFO.                 │ ▒     │
│                                         │ ▒     │
│       ▦  TEMPERATURE INFO.              │ ▒     │
│                                         │ ▒     │
│       ▦  EQUIPMENT INFO.                │ ▒     │
│                                         │ ▒     │
│       ▦  BUILDING INFO.                 │ ▒     │
│                                         │ ▒     │
│       ▦  SPEC. INFO.                    │ ▒     │
│                                         │ ▒     │
│                                         │ ▒     │
│                                         │ ▒     │
│                                         │ ▼     │
├───┬─────────────────────────────────┬───┴───────┤
│ ◄ │                                 │ ► │        │
└───┴─────────────────────────────────┴───────────┘
```
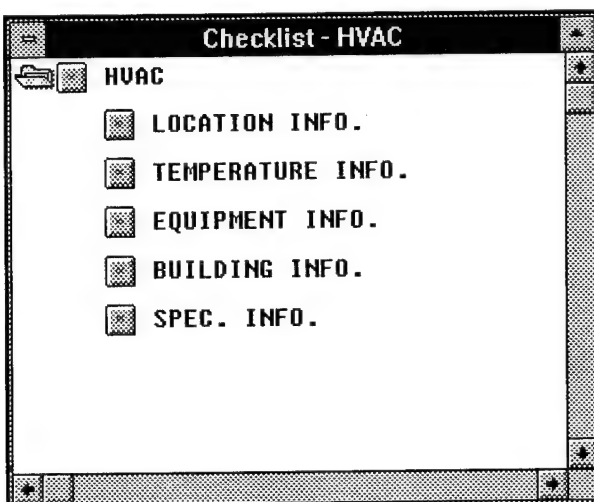
There are two ways to execute a checklist item: (1) select a specific checklist item to execute or (2) execute the first item that has not been completed by selecting the top checklist item. A checklist is considered complete when all of the required checklist items are complete.

To execute a process in the Process View

1. Expand the *process tree* to display the process you want to execute.

2. Choose the desired process by clicking its button.

   The process will search for the first agent that has not completed its tasks. The first checklist within that agent that has not been completed will be shown in the Checklist window. If all checklists have been completed, a message to that effect will be displayed and the Checklist window will not change.

3. Execute the checklist.

   See "To execute a checklist in the Process View" on page 23.

4. Repeat steps 2 and 3 until all checklists in the process have been completed.

To execute an agent in the Process View

1. Expand the process that contains the agent you want to execute.

2. Choose the agent you want to execute by clicking its button.

   A checklist will be displayed in the Checklist window. The first checklist that has not been completed will be shown. If all of the agent's checklists have been completed, a message will be displayed and the Checklist window will not change.

3. Execute the checklist.

   See "To execute a checklist in the Process View" on page 23.

4. Repeat steps 2 and 3 until all checklists in the agent have been completed.

To display a checklist in the Process View

1. Expand the process to reveal the agent you want.

2. Expand the agent to show its checklists.

3. Choose the desired checklist by clicking its button.

    The checklist will be displayed in the Checklist window.

To execute a checklist in the Process View

1. Display the checklist in the Checklist window.

2. Click on the top checklist button in the Checklist window.

    The first required item that is incomplete will execute. When a checklist item is complete, the checklist item button is marked with an X.

3. Repeat step 2, or click on the next incomplete item button, until all desired items are completed.

To execute a checklist item in the Process View

1. Display the checklist in the Checklist window.

2. Click on the checklist item button.

    The item will then execute. It may display a message, ask for more information, or run an external program. When a checklist item is complete, the checklist item button is marked with an X.

Summary of Process View Options

The Process View allows you to:

- view the flow and progress of processes in your project

- add processes to your project

- execute checklists and view the results.

The menu bar contains File, Process, View, Window, and Help functions. File is for working with projects. Process allows you to add and delete processes to your project. View allows you to switch to the Agent View or the Instance View. Window allows you to activate the Developer window and the LISP window, and organize the open windows on the computer screen. Help contains the Ace online help facility.

Within the Process window, the agents, checklists, and instances are displayed in a hierarchical tree structure. Within the Checklist window, checklists are displayed and executed.

Process View Menu Options

File

New	Initiates a new project.
Open	Opens an existing project.
Close	Closes the current project.
Save	Saves the current project.
Save As	Saves the current project under a new name.
Delete	Deletes a selected project.
Properties	Displays and allows modification of project information for the current project.
Exit	Exits Ace.

Process

Add	Adds a process to the current project.
Remove	Removes a process from the current project.

View

Processes	Accesses the Process View.
Agents	Accesses the Agent View.
Instances	Accesses the Instance View.

Window

Project	Activates the Project window.
Developer	Opens and activates the Developer window.
Cascade	Arranges all windows to overlap, with each window title bar visible.
Tile	Changes window sizes so all active windows are equal sizes and arranged neatly on screen.
LISP	Opens and activates the LISP window.
Messages	Opens and activates the Message window.

Help

Contents	Displays a list of Help topics available for Ace.
Search for Help on...	Displays a dialog box for locating information about a specific Ace topic.
How to Use Help	Displays guidance on using Help.
Process View Help	Displays the "Using the Process View" topic in Help.

Process Window Options

▦	Process button	This button displays in the Checklist window the first incomplete checklist in the process.
▦	Agent button	This button displays in the Checklist window the first incomplete checklist in the agent.
▦	Checklist button	This button displays in the Checklist window the checklist whose name appears beside the button.
▭	Closed folder icon	Clicking on this icon will expand the tree display to show the folder contents.
▱	Open folder icon	Clicking on this icon will collapse the display tree to hide the folder contents.

Checklist Window Options

▦	Checklist Item button in *up* state	This button represents a checklist item that has not been completed. Pressing the button executes the checklist item.
▦	Checklist Item button in *down* state	This button represents a checklist item currently being executed.
▨	Checklist Item button in *up* state (for completed item)	Designated with an X, this button represents a checklist item that has been completed. Pressing the button will execute the checklist item again.
▨	Checklist Item button in *down* state (for completed item)	Designated with an X, this button represents the completed checklist item that is currently being executed again.
▭	Closed folder icon	Clicking this icon will expand the checklist display to show the sub-items.
▱	Open folder icon	Clicking this icon will collapse the tree display to hide the sub-items.

Chapter 5: The Agent View

Using the Agent View

The Agent View is used to examine agents and their components. Unlike the Process View, there is no concept of "what to do next" in the Agent View. Here, the user is allowed to browse through the agents that have been added to the project and interact with their checklists and instances. Users can add and remove agents from the project in the same way they can add and remove processes in the Process View.

During an Ace session, the user may create new instances and define relationships between instances in the Instance View. To represent these actions, Ace provides a special-purpose agent called the *User* agent, which is visible in the Agent View.

Through the Agent View, you can:

- add agents to a project

- remove agents from a project

- execute an agent's checklists

- access the instances created by an agent.

For an overview of agents, see "Agents" on page 9. For information on agents in the Process View, see "Using the Process View" on page 19. For information on creating new agents, refer to the <u>Ace Developer's Manual</u> or the Help menu in the Developer window.

The Agent View contains the following:

- The *Project title bar* displays "Ace" and the name of the active project.

- The *Agent View menu bar* contains the available menus that can be opened to choose commands.

- The *Agents window* displays agents and their checklists and instances.

- The *Components window* displays one of the following: a list of the agents in a project, a list of checklists and instances in a specific agent, a specific checklist, or a specific instance.

- The *Message window* displays messages about commands being performed.

- The *Status line* displays the name of the window that is currently under the mouse pointer.

Agent Window ——————
Agent Tree ——————
Components Window ——————
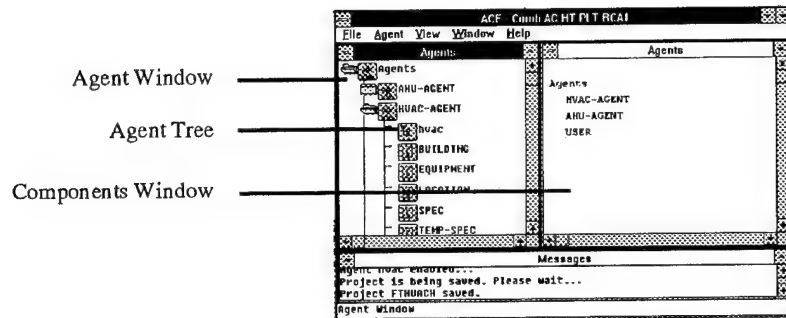
For a quick reference to the available options, see "Summary of Agent View Options" on page 32.

## To access the Agent View

- Choose Agents from the View menu.

  The Agent View will appear in the Project window.

Viewing and Modifying the Agent Tree

The Agents window displays the agent tree containing the agents in your project. Agents are all shown on the same hierarchical level of the agent tree, and checklists and instances are listed under their agent. Agents can be added or removed from the project here. An agent's checklists can be displayed, and the instances created by an agent can be accessed.

The agent tree provides icons that are used to expand and collapse the tree, and includes buttons for displaying checklists and instances.

- Selecting *folder icons expands* the tree to show checklists and instances. Clicking on a closed folder displays the next level of the tree. Clicking on an opened folder collapses that level.

- Clicking *agent buttons* enables you to view items on the agent tree in a list format. Clicking on the top agent button (labeled Agents) lists the project's agents in the Components window. The agent button for a specific agent will list that agent's checklists and instances.

- Clicking *checklist buttons* displays checklist items in the Components window. To execute the checklist, see "Executing Checklists in the Agent View" on page 30.

- Clicking *instance buttons* allow you to view instances in the Components window. To modify Instances, see "Viewing and Modifying Instances in the Agent View" on page 31.
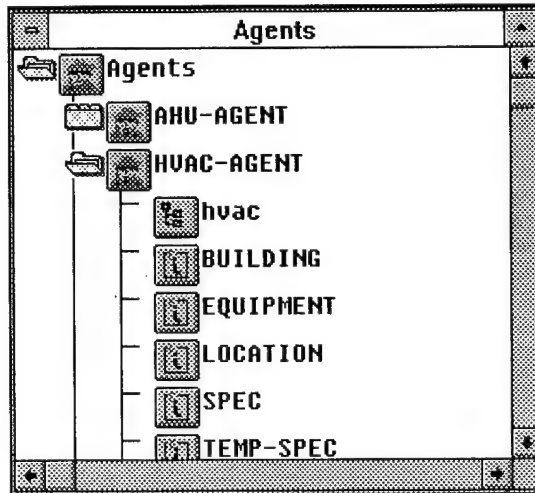
To display the components of the agent tree

The folder icons can be used to either display or hide items on the agent tree. A folder icon will appear open if an item is expanded and closed if an item is collapsed. An item without a folder icon next to it does not have any items below it in the hierarchy.

- Clicking on the folder icon next to the button labeled Agents, at the top of the tree, displays or hides the agents names in the project.

- Clicking on the folder icon next to an agent button displays or hides the agent's checklists and instances names.

To add an agent to a project

Agents are added to a project in two ways. When a process is added to a project through the process view, the agents it uses are automatically added to the project. Agents can also be added to a project through the Agent View. Before adding an agent to the project it must be installed in the system. See "Installing and uninstalling Agents" on page 17.

1. Choose Add from the Agent menu.

   A dialog box will appear. If the agent you want is not listed in the box, it needs to be installed.

2. Choose the agent you want to add.

   The agent will be added to the agent tree.

   Note: Only agents currently installed in Ace will appear in the dialog box. For information on installing agents, refer to the <u>Ace Developer's Manual</u> or the Help menu in the Developer window.

To remove an agent from a project

1. Choose Remove from the Agent menu.

   A dialog box will appear containing a list of the agents in the project.

2. Select the agent you want to remove.

   The agent will be deleted from the agent tree.

To list a project's agents in the Agent View

For projects that contain a large number of agents it sometimes is useful to view the agents in a list format, without buttons and folders.

- Click the button labeled Agents, at the top of the agent tree.

  A list of the agents in the project will be displayed in the Components window. When clicked with the mouse, each item on this list functions the same way as the corresponding agent button in the agent tree.

To list an agent's checklists and instances in the Agent View

For agents that contain a large number of checklists and instances, it is sometimes useful to view the components in a list format, without buttons and folders.

1. When necessary, expand the agent tree by clicking on the folder icon next to the button labeled Agents.

2. Choose the desired agent by clicking its button.

   A list of the agent's checklists and instances will be displayed in the Components window.

Executing Checklists in the Agent View

Checklists are displayed in the Components window as a hierarchical tree of items. Each item can be executed to supply a specific piece of information or perform a certain function. The button of each checklist item will be marked with an X when completed. For an overview of checklists, see "Checklists" on page 10.

A checklist is considered completed when all of the items within it are completed. There are two ways to execute checklist items: (1) selecting a specific checklist item to execute or (2) executing the first incomplete checklist item by selecting the top checklist item.

To display a checklist in the Agent View

1. When necessary, expand the agent to reveal its checklists by clicking the folder icon next to the agent button.

   The checklists will be listed both in the Agent window and in the Components window.

2. Choose a specific checklist by clicking its button in the Agent window or by clicking on the checklist name in the Components window.

   Checklist items will then be displayed in the Components window.

To execute a specific checklist item in the Agent View

1. Display the checklist items in the Components window.

2. Choose a specific checklist item by clicking on the item's button.

   The item will then execute. It may display a message, ask for more information, or run an external program. A checklist item is complete when the checklist item button is marked with an X.

Checklist - HVAC

- HVAC
  - LOCATION INFO.
  - TEMPERATURE INFO.
  - EQUIPMENT INFO.
  - BUILDING INFO.
  - SPEC. INFO.
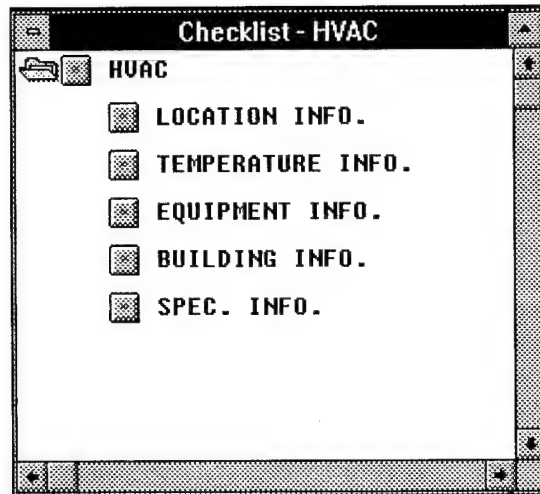
To execute the first incomplete checklist item in the Agent View

1. Display the checklist items in the Components window.

2. Click on the top checklist item.

   The first checklist item that has not been completed will execute. It may display a message, ask for more information, or run an external program. A checklist item is complete when the checklist item button is marked with an X.

Viewing and Modifying Instances in the Agent View

Instances created by an agent are shown on the agent tree, below the agent. These instances can be displayed and modified in the Agent View by clicking the associated instance button on the agent tree. When the instance is displayed, it can be modified in the same manner as in the Instance View. For more information on modifying instances, see ""Displaying and Modifying Instance Slot Data" on page 42.

To display an instance in the Agent View

1. When necessary, expand the agent to reveal its instances by clicking the folder icon next to the agent button.

2. Choose the desired instance by clicking its button in the Agent window or by clicking its name in the Components window.

   The instance will be displayed in the Components window, where it can be viewed and modified.

To modify instance data in the Agent View

1. Display the instance in the Components window.

2. Click the slot value that you want to change, then enter the new value.

   If the slot value is asterisks, click twice. The asterisks indicate the value of the slot is nil.

Summary of Agent View Options

The Agent View allows you to view the agents in your project, modify those agents, execute checklists, and view and modify instances.

The menu bar contains File, Agent, View, Window, and Help functions. File is for working with projects. Agent allows you to add and delete agents to your project. View allows you to switch to the Process View or the Instance View. Window enables you to switch to the Developer window and the LISP window, and to organize the open windows on screen. Help provides access to the Ace online help facility.

Within the Agents window, agents, checklists, and instances are displayed in a tree structure. Within the Components window checklists are displayed and executed, and instances are viewed and modified.

Agent View Menu Options

File

New	Initiates a new project.
Open	Opens an existing project.
Close	Closes the current project.
Save	Saves the current project.
Save As	Saves the current project under a new name.
Delete	Deletes a selected project.
Properties	Displays and allows modification of the project information for the current project.
Exit	Exits Ace.

Agent

Add	Adds an agent to the current project.
Remove	Removes an agent from the current project.

View

Processes	Accesses the Process View.
Agents	Accesses the Agent View.
Instances	Accesses the Instance View.

Window

Project	Activates the Project window.
Developer	Opens and activates the Developer window.
Cascade	Arranges all windows to overlap, with each window title bar visible.
Tile	Changes window sizes so all active windows are equal sizes and arranged neatly on screen..
LISP	Opens and activates the LISP window.
Messages	Opens and activates the Message window.

Window

Project	Activates the Project window.
Developer	Opens and activates the Developer window.
Cascade	Arranges all windows to overlap, with each window title bar visible.
Tile	Changes window sizes so all active windows are equal sizes and arranged neatly on screen.
LISP	Opens and activates the LISP window.
Messages	Opens and activates the Message window.

Help

Contents	Displays a list of Help topics available for Ace.
Search for Help on...	Displays a dialog box for locating information about a specific Ace topic.
How to Use Help	Displays guidance on using Help.
Agent View Help	Displays the "Using the Agent View" topic in Help .

Agent Window Options

The Agent window contains the agent tree which is a display of the agents in your project.

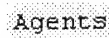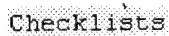|		
	Agent button	Clicking the top button labeled Agents lists the project's agents in the Components window. Clicking the button for a specific agent lists the agent's checklists and instances in the Components window.
	Checklist button	Clicking this button displays the corresponding checklist in the Components window.
	Instance button	Clicking this button allows the corresponding instance to be viewed and modified in the Components window.
	Closed folder icon	Clicking on this icon will expand the tree display to show the folder contents.
	Open folder icon	Clicking on this icon will collapse the tree display to hide the folder contents.

Components Window Options

The Components window can display lists of agents, checklists and instances, a checklist's items, or an instance's slots and values (depending on which actions have been performed).

Agent List Options

Agents	This the heading for the list of agents in the current project.
<agent name>	The agent name, when clicked, lists the agent's checklists and instances in the Components window.

Checklist and Instance List Options

Checklists	This is the heading for the list of checklists belonging to a specific agent.
<checklist name>	The checklist name, when clicked, displays the corresponding checklist in the Components window, where it then can be executed.
Instances	This is the heading for the list of instances belonging to a specific agent.
<instance name>	The instance name, when clicked, displays the instance in the Components window, where it then can be viewed and modified.

Checklist View Options

▨	Checklist Item button in *up* state	This button represents a checklist item that has not been completed. Pressing the button executes the checklist item.
▨	Checklist Item button in *down* state	This button represents a checklist item currently being executed.
▨	Checklist Item button in *up* state (for completed item)	Designated with an X, this button represents a checklist item that has been completed. Pressing the button will execute the checklist item again.
▨	Checklist Item button in *down* state (for completed item)	Designated with an X, this button represents the completed checklist item that is currently being executed again.
▤	Closed folder icon	Clicking this icon will expand the checklist display to show the sub-items.
▤	Open folder icon	Clicking this icon will collapse the tree display to hide the sub-items.

Chapter 6: The Instance View
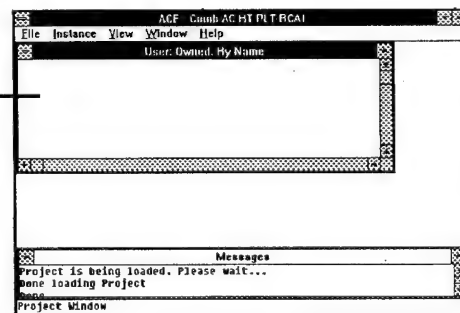
Using the Instance View

The project data are stored in *instances*. The Instance View provides complete access to the instances in a project. Unlike the Agent View, you can view and modify instances independent of their agents. Custom viewpoints of the data can be created as an instance set. Each viewpoint has two primary aspects: which instances are displayed and how they are arranged. The names of the instances in the set are displayed in an instance set window. An instance can be selected to be viewed or modified, added or deleted, and linked with other instances.

More than one instance set window can be open at a time, and it can be informative to compare different views of the instances. For example, you might want to compare instances that were created by your pump agent with the instances of pumps regardless of where they were created.

An instance set window can be deleted when it is no longer needed, or it can be closed temporarily. Some project changes may not be shown immediately in the current instance set windows. The windows can be refreshed at any time to reflect the latest changes by choosing Refresh from the View menu..

Each instance is created by an agent from a *frame*. An instance is considered to be *owned* by the agent that created it. Instances created in the Instance View are owned by the *User Agent*. For an overview of frames and instances, see "Frames and Instances" on page 11.

Instance Set Window —————

The Instance View contains the following:

- The *Project title bar* displays "Ace" and the name of the active project.

- The *Instance View menu bar* contains the available menus that can be opened to choose commands.

- An *Instance set window* displays names of instances that have been selected by common characteristics.

- The *Message window* displays messages about commands being performed.

- The *Status line* displays the name of the window that is currently under the mouse pointer.

For a quick reference to the available options, see"Summary of Instance View Options" on page 46.

To access the Instance View

- Choose Instances from the View menu.

  The Instance View will appear in the Projects window. Within the Instance View an instance set window will display a list of the instances owned by the User Agent.

Instance Sets in the Instance View

Within the Instance View, instance set windows are used to display the names of selected sets of instances. An instance is selected from an instance set window to be examined, modified, linked to another instance, or deleted. To quickly view and modify any specific instance, see "To display an instance using the Find option" on page 42.

When selecting a set of instances, you can choose from two viewpoints: the Agent viewpoint or the Query viewpoint.

The *Agent viewpoint* shows the instances of a specific agent. You can select the agent's owned instances or the agent's interest set.

- *Owned* instances are those created by the agent, and are considered to be owned by that agent.

- The *interest set* contains the instances the agent owns plus any in which it has an interest. The agent is considered to be interested in any instance that it must access to do its job, whether the agent receives information from the instance or provides information to it. The interest set is specified when the agent is created. Because the User Agent represents you as you interact with your project, you specify the User Agent's interest set when selecting a set of instances. Once selected, these instances remain in the User Agent's interest set.

The *Query viewpoint* allows you to look at any combination of instances from specific frames. You can choose one frame, more than one frame, or all of the frames used in the project.

For both viewpoints, when you select the instances, you can arrange the display in three ways.

- The *frame display* lists frames, with their instances listed under them in a tree structure.

- The *instance name display* lists the instances in alphabetical order.

- The *semantic link display* shows the relationship between instances. The successor instance in the link is indented under the predecessor instance. For more information on semantic links, see "Semantic Links for Instances" on page 41.

Whenever an instance set is displayed with frames, a folder icon and a graphic box are displayed beside each frame name. Clicking on the folder icon opens the folder. The frames and instances inside it are displayed on the next level. Clicking on an open folder closes it and hides its contents. The graphic box indicates whether the frame contains child frames and instances.

This box indicates that the frame contains child frames and instan

This box indicates that the frame contains instances only.

In the following figure, one instance set is displayed. It is the agent viewpoint for the HVAC agent. Any instance affecting or affected by the HVAC agent is displayed. The instances are shown with their frames. According to the graphic boxes next to each frame, none of the frames contains any child frames.


```
ACE - Comb AC HT PLT-BCA1
File   Instance   View   Window   Help
                          hvac: Interest: By Frame
   BB-OBJECT
      BUILDING-FRAME
         └    BUILDING
      EQUIPMENT-FRAME
      LOCATION-FRAME
      SPEC-FRAME
      TEMP-SPEC-FRAME
      TEMPERATURES-FRAME




Interest Set of Agent hvac showing By Frame
```
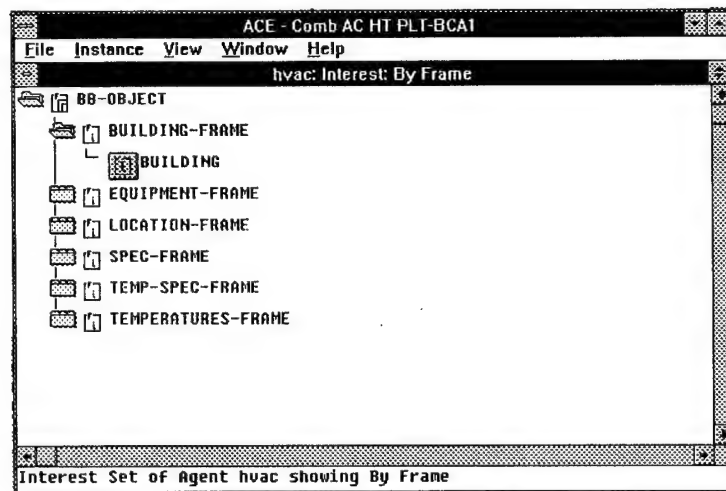
To select a set of instances from a specific agent

1. Access the Instance View.

2. Choose New from the Window menu.

3. Choose :Agent.

4. Select the name of the agent whose set of instances you want to see.

5. Select whether you want to see the agent's Interest Set or Owned instances.

6. Select whether you want the instances arranged By Frame, By Name or by a selected link type.

7. If you select the User Agent and its interest set, you must select the frames to be included. These frames become the User Agent's interest set. Subsequent selections are added to the interest set as follows:

   • A list of frames will be presented one level at a time.

   • At each level, select any desired frames.

   • Step through each frame's levels to select its child frames.

   • Choose All from the menu at any time to select all frames displayed at that level, and all of their child frames.

   An instance set window will display the names of the instances that match the selections.

To select a set of instances from specific frames

1. Access the Instance View.

1. Choose New from the Window menu.

2. Choose :Query.

3. Select whether you want the instances arranged By Frame, By Name, or by a selected link type.

4. Select the frames of the instances you wish to see.

   • A list of frames is presented one level at a time.

   • At each level, select any desired frames.

   • Step through each frame's levels to select its child frames.

   • Choose All from the menu at any time to select all frames displayed at that level, and all of their child frames.

   An instance set window will display the names of the instances that match the selections.

To activate an open instance set window

• Click on the instance set window within the Instance View.

To close an instance set window in the Instance View

1. Activate the instance set window you wish to close.

2. Choose Close from the Window menu.

   The window is removed from the display, but can be opened again when it is needed.

To open a closed instance set window in the Instance View

• From the Window menu, click on the name of the instance set window you want to open.

The window will open and become active.

To delete an instance set window

1. Activate the instance set window you wish to delete.

2. Choose Delete from the Window menu.

   The instance set window is deleted and is no longer accessible.

To modify the list of instances in an instance set

1. Activate the instance set window you wish to change.

2. Choose Customize from the Window menu.

   A pop-up window displays the properties for that set of instances.

3. Click twice on the property you wish to change.

4. Select the value you want.

5. Repeat Steps 3 and 4 until you have changed the properties to your desired values.

6. Click OK when finished.

   The instance set window will change to reflect the new properties.

To refresh the instance set windows

- In the Instance View, choose Refresh from the View menu.

  All of the instance set windows will be updated to reflect any changes made since the previous refresh.

Adding, Annotating, and Deleting Instances

New instances can be created from any frame available to your project. Frames are created in the Developer window. For information on creating new frames, refer to the Ace Developer's Manual or the Help menu in the Developer window. Instances created in the Instance View are created by you, the user. As such, these instances will be owned by the User Agent and are included in the interest set of any agents in the project.

Instances also can be deleted. *Care should be taken when deleting instances because other agents may depend on them.* You must fully consider how and where the instance is used in the project before deleting it.

Any instance can be annotated with documentation about the instance. More than one annotation can be created for the instance. After the annotation has been added, it can be viewed, edited, or deleted.

To create an instance

1. Choose New from the Instance menu.

2. Select the frame for the type of instance you want to create.

3. Enter a unique name for the new instance.

To delete an instance

1. From an instance set window, click on the name of the instance to be deleted.

   For information on selecting a set of instances in an instance set window, see "Instance Sets in the Instance View" on page 36.

2. Choose Delete from the Instance menu.

3. Confirm the deletion.

To create, display, or modify an annotation of an instance

1. Select an instance from an instance set window.

2. Choose Annotation from the Instance menu.

   The Annotation Options window will be displayed.

3. Choose Add, View, or Edit.

4. For View or Edit, select the annotation.

   A file, annotate.txt, will be opened in Notepad, the Windows® text editor. If you select View, you will not be able to edit this file.

5. Type new annotation or any changes at the top of the file.

6. For Add or Edit, choose Save from the File menu in Notepad.

7. Exit Notepad.

8. To confirm your addition or edit, click on OK in the confirmation window.

To delete an annotation of an instance

1. Select the instance from an instance set window.

2. Choose Annotation from the Instance menu.

   The Annotation Options window will be displayed.

4. Choose Delete.

5. Select the annotation to delete.

   It will be deleted.

Executing Instance Rules for a Project

Some projects will have rules associated with certain frames or instance slots. If a new instance has been created that might be associated with a rule, that rule should be executed. Similarly, if an instance is deleted, any rules associated with it should be run again. For more information about agent rules, refer to the Ace Developer's Manual or the Help menu in the Ace Developer window.

To execute the rules associated with an instance

- Choose Run Rules from the Instance menu.

  All rules active in the project will be executed.

Semantic Links for Instances

Relationships can be created between instances. These relationships are called *semantic links*. There are different types of semantic links. For example, there might be a *has-part* link between a pump and its impeller, or a *supported-by* link between a beam and a column. Several standard link types are provided with Ace, but you can also create your own types.

In an instance set window arranged by link type, the instances will be displayed in a tree structure with the successor listed under the predecessor. The predecessor is the first instance selected when the link was created, and the successor is the second instance selected. The semantic link between instances is not displayed in instance set windows arranged by frame or instance name. When working with semantic links, it is easiest to use a window that displays the links.

To link instances

Each of the instances you want to link must be in the same instance set window. For information on selecting a set of instances in an instance set window, see "Instance Sets in the Instance View" on page 36.

1. From an instance set window, click on the name of the predecessor instance.

2. Choose Semantic Link from the Instance menu.

3. Choose Link objects.

4. Click on the successor instance.

5. Select the type of link.

   The link will be displayed when you have an instance set window arranged by that link type.

To unlink instances

The predecessor instance must be displayed in an instance set window. For information on selecting a set of instances in an instance set window, see "Instance Sets in the Instance View" on page 36.

1. From an instance set window, click on the name of the predecessor instance containing the link.

2. Choose Semantic Link from the Instance menu.

3. Choose Unlink objects.

4. Click on the successor instance to be unlinked.

To create a new semantic link type for your project

1. Choose Semantic Link from the Instance menu.

2. Choose Create link type.

3. Enter a unique name for the link type.

4. Enter a unique print name for the link type.

   The print name is used when the link is displayed.

To delete a semantic link type in your project

1. Choose Semantic Link from the Instance menu.

2. Choose Delete link type.

3. Select the link type you wish to delete.

   The link will be deleted.

Displaying an Instance in the Instance View

Instances can be displayed either by using the Find option to access a specific instance, or by choosing to edit an instance listed in an instance set window. When an instance is displayed, the data can be modified, and the slot rationales and constraints can be accessed. For more information on modifying instances, see "Displaying and Modifying the Instance Slot Data" on page **Error! Bookmark not defined.**.

To display an instance using the Find option

1. Choose the Find option from the Instance menu.

   A list appears containing the names of all instances in the project.

2. Select the instance you want to display.

   The frame and instance names will be displayed with the instance.

To display an instance from an instance set window

1. From an instance set window, click on the name of the instance to be displayed.

   For information on selecting a set of instances in an instance set window, see "Instance Sets in the Instance View" on page 36.

2. Choose Edit from the Instance menu.

   The instance will be displayed.

Displaying and Modifying Instance Slot Data

There are four components of an instance slot that can be accessed: value, annotation, constraint, and rationale. Slot names and values are displayed when the instance is displayed; the other three components are accessible through an options menu.

1. The *slot value* is the project data itself. You can view the slot values and modify them by adding values, changing existing values, or deleting values.

2. An slot value *annotation* is a comment about the slot value. More than one annotation can be made to a the value. After the annotation has been added, it can be viewed, edited, or deleted.

3. A *constraint* is a user-specified boundary or scope for the slot value. Constraints can be ignored, restored, or removed. You can view its history or the rationale behind it.

4. The slot value *rationale* documents how the value was chosen. The text states whether the value is a default or an axiom. If the rationale arises from a rule, the rule will be displayed.
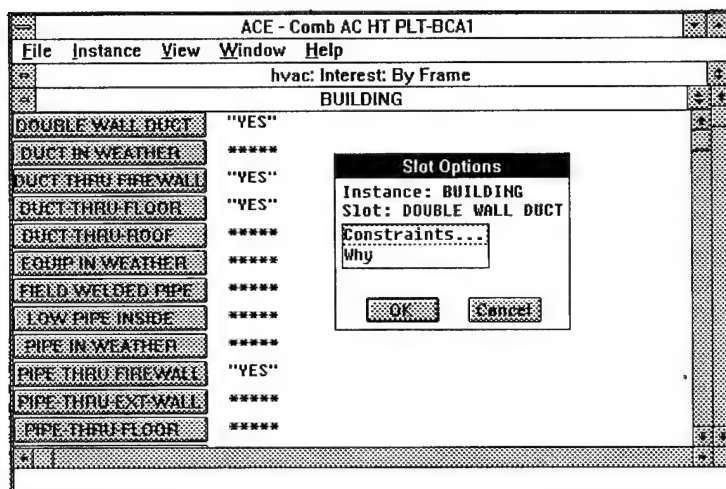
To modify the value of an instance slot in the Instance View

1. Display the instance that contains the slot to be modified.

2. Click on the slot value that you want to change and enter a new value.

   If the slot value is asterisks, click twice. The asterisks indicate the value of the slot is nil.

To create, display, or modify an instance slot value annotation

1. Display the instance.

2. Click on the slot name.

   The Slot Options popup menu will be displayed.

3. Choose Annotation.

   The Annotation Options popup menu appears.

4. Choose Add, View, or Edit.

5. For View or Edit, select the annotation.

   A file called annotate.txt will be opened by the Windows®
   Notepad utility. If you selected View, you will not be able to edit
   the file.

6. Type a new annotation or edit the existing one.

7. When you finish creating or editing the annotation, choose Save
   from the File menu in Notepad.

8. Exit Notepad.

9. Click OK when prompted in the confirmation box.

To delete an instance slot value annotation

1. Display the instance.

2. Click on the slot name.

   The Slot Options popup menu appears.

3. Choose Annotation.

   The Annotation Options popup menu appears.

4. Choose Delete.

5. Select the annotation to be deleted.

   It will be deleted.

To create an instance slot value constraint

1. Display the instance.

2. Click on the slot name.

   The Slot Options popup menu appears.

3. Choose Constraints.

   The Constraints Options popup menu appears.

4. Choose Add a Constraint.

5. Select the type of constraint.

6. Select whether each variable in the constraint type is a *slot value*
   or an *explicit value*.

   With a slot value type, the constraint is related to the value of
   another instance slot. This value can vary during the execution of
   the agent. An explicit value type is one the user supplies, and it
   will not change.

7. If the variable is a slot value, select the frame and then the slot. If
   the variable is an explicit value, enter the value.

8. Enter a rationale annotation.

To display an instance slot value constraint

1. Display the instance that contains the slot to be viewed.
2. Click on the slot name.

   The Slot Options popup menu appears.
3. Choose Constraints.

   The Constraints Options popup menu appears.
4. Choose View a Constraint.

   If there are any constraints for the selected slot, they will be listed. You can select one to obtain more information about it.

To display an instance slot value constraint history or rationale

1. Display the instance to be viewed.
2. Click on the slot name.

   The Slot Options popup menu appears.
3. Choose Constraints.

   The Constraints Options popup menu appears.
4. Select View History or Show Rationale.

   The constraint history will be displayed in a popup window. The constraint rationale will appear in the Message window.

To ignore, restore, or delete an instance slot value constraint

1. Display the instance you want.
2. Click on the slot name.

   The Slot Options popup menu appears.
3. Choose Constraints.

   The Constraint Options popup menu appears.
4. Select Ignore a Constraint, or Restore a Constraint, or Remove a Constraint.

To display an instance slot value rationale

1. Display the instance that contains the slot to be viewed.
2. Click on the slot name.

   The Slot Options popup menu appears.
3. Choose Why.

   The rationale text will appear in the Message window.

Summary of Instance View Options

The Instance View allows you to access instances in your project independently of your agents. You can add, view, modify, delete instances, and you can create links between them.

The Instance View menu bar contains File, Instance, View, Window, and Help. File is for working with projects. Instance is for working with instances when a set of instances is displayed in a instance set window. View is for switching between the Instance View. and other views. Window is for switching to the Developer window or the LISP window, organizing open windows on screen, and creating instance set windows. Help contains the Ace online help.

When selecting a set of instances to appear in an instance set window, functional options are available for the instance viewpoint, the arrangement of instances, and the display types of an agent.

Instance View Menu Options

File

New	Initiates a new project.
Open	Opens an existing project.
Close	Closes the current project.
Save	Saves changes made in the current project.
Save As	Saves the current project under a new filename.
Delete	Deletes a selected project.
Properties	Displays and allows modification of the project information for the current project.
Exit	Exits Ace.

Instance

New...	Creates a new instance.
Edit..	Displays and allows modification of the selected instance displayed in an instance set window.
Annotate...	Adds, views, edits, or deletes an annotation of an instance.
Delete	Deletes the selected instance in an instance set window.
Find	Displays an instance selected by name.
Semantic Link	Menu for working with semantic links.

Link objects	Creates a semantic link.
Unlink objects	Deletes the selected link.
Create link type	Creates a new link type.
Delete link type	Deletes a type of link.

Run Rules	Executes the rules associated with the current project.

View

Processes	Accesses the Process View.
Agents	Accesses the Agent View.
Instances	Accesses the Instance View.
Refresh	Updates all windows displayed in the Instance View.

Window

Project	Activates the Project window.
Developer	Opens and activates the Developer window.
Cascade	Arranges all windows to overlap, with each window title bar visible.
Tile	Changes window sizes so all active windows are equal sizes and arranged neatly on screen.
New	Creates a new instance set window.
Customize	Displays and allows modification of the attributes of the active instance set window.
Close	Closes the active instance set window.
Delete	Deletes the active window.
LISP	Opens and activates the LISP window.
Messages	Opens and activates the Message window.
<Instance Set window>	The current Instance Set windows are listed here. When one is selected, the window is opened.

Help

Contents	Displays a list of Help topics available for Ace.
Search for Help on...	Displays a dialog box for locating information about a specific Ace topic.
How to Use Help	Displays guidance on using Help.
Instance View Help	Instance View Help.

Instance Set Selection Options

Viewpoint Options

:Agent	Selects the instances of a specific agent.
:Query	Selects the instances of one or more frames.

Arrange Types Options

By Frame	Orders instances by frame.
By Name	Orders instances alphabetically by name.
<Link type>	Orders instances by selected link type.

Agent Display Types Options

Interest Set	Selects instances that are in the agent's interest set.
Owned	Selects instances that were created by the agent.

USACERL DISTRIBUTION

Chief of Engineers
   ATTN: CEHEC-IM-LH (2)
   ATTN: CEHEC-IM-LP (2)
   ATTN: CECC-R
   ATTN: CERD-L

Directorate of Military Programs
   ATTN: CEMP-ET

Defense Tech Info Center 22304
   ATTN: DTIC-O (2)

<div align="right">

9
+10
12/95

</div>